CENTER FOR CYBERNETIC STUDIES

The University of Texas
Austin, Texas 78712

Research Report CCS 195

IMPORTANT PRACTICAL MISCONCEPTIONS OF OPTIMIZING LARGE SCALE ASSIGNMENT AND TRANSPORTATION PROBLEMS

by

F. Glover*
D. Klingman

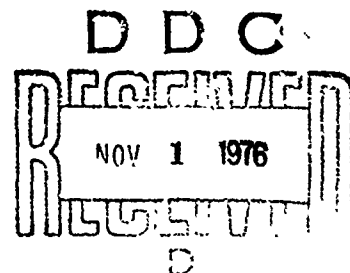October 1974

* University of Colorado, Boulder, Colorado 80302.

D D C
RECEIVED
NOV 1 1976
RECEIVED
D

CENTER FOR CYBERNETIC STUDIES

A. Charnes, Director
Business-Economics Building, 203E
The University of Texas
Austin, Texas 78712
(512)   471-1821

INTRODUCTION

Modelling and solving large scale networks are crucial to many practical military applications. The purpose of this note is to identify important elements of successful models and methods that were incompletely or inaccurately portrayed in recent presentations at O.R. meetings such as that in [16]. Our observations result from breakthroughs in network areas that have been validated in over a hundred computer hours of empirical testing and 15 man years of code development. They particularly apply to modelling large scale military manpower assignment problems and designing computer codes for solving large scale assignment, transportation, and transshipment problems. To focus our remarks we shall address chiefly the misconceptions presented in [16].

MEMORY AND SOLUTION CAPACITIES

A major oversight of [16], which unfortunately is transmitted throughout the paper, concerns a confusion between computer codes for capacitated and uncapacitated network problems (and a secondary confusion between early codes and more recent ones). One manifestation of this confusion occurs in the formula given in [16] for computing memory requirements for the recently developed network code PNET [7]:

$$3 \ (mnd) + 8m + 7n + 10,000$$

where

m + number of source nodes

n = number of sink nodes

d = cost matrix density

This is not the formula for PNET, but is the formula for the somewhat earlier transportation code PTRANS [9], and applies to a version for solving <u>capacitated</u>

transportation problems. The correct formula for PNET is

$$2(mnd) + 5m + 5n + 8,000$$

The errors resulting from this misunderstanding cause the entries in Table 1
and Figure 1 to be drastically distorted. While it appears in Table 1 and
Figure 1 that PNET can not solve as large a problem as the other in-core
codes listed, PNET can in fact solve larger problems than any other in-core
codes in existence. In addition, PNET is capable of solving general trans-
shipment problems as well as assignment and transportation problems. None
of the other codes discussed in [16] has this ability. (Ironically, the
difficulties of optimal quota accommodation--for "fill" optimization--discussed
at length in [16] are in fact due to this inability of the other codes to
solve transshipment problems. PNET's ability to handle such problems eliminates
the need for a nonlinear optimization routine.)

Our research over the past five years has solidly demonstrated that
simplex-based computer codes are more efficient and require less memory than
primal-dual (out-of-kilter) computer codes. This empirical fact has been
derived scientifically by developing and implementing a wide variety of
improved algorithmic procedures for network [1, 3, 7, 9, 10, 15, 17, 18, 19, 23].
(Our findings concerning the superiority of simplex-based codes are not biased
by inattention to primal-dual methods. In fact, our primal-dual code SUPERK [1]
has never been beaten by any other primal-dual code.)

To verify the practical merit of these developmental efforts, we have
conducted extensive computational testing (in excess of 100 central processing
hours on a CDC 6600) against all available codes and on all types of assignment,
transportation, and transshipment problems [7, 9, 10, 15, 17, 18, 19, 23].

The outcomes of this testing were then validated across different types and sizes of computer; e.g., CDC 3100, UNIVAC 1108, Burroughs 4700, CDC 6600, IBM 360/65, IBM 370/155, CDC 6400, Burroughs 6700, PDP-10, and IBM 370/145. Subsequently, the efficiency of our codes and the accuracy of our conclusions have been independently substantiated by researchers around the world.*

These developments uncover a serious omission in the hypothetical codes considered in [16], which astonishingly fail to include simplex-based codes. The eligibility and cost storage schemes mentioned in [16] are all easily accommodated by a simplex-based code. As a consequence, a simplex-based network code can be designed whose memory requirement is only 2n + 2m words beyond that required by the cost storage scheme. This memory requirement is less than any of the hypothetical or existing codes discussed in [16].

We are quite skeptical of the value of in-core codes utilizing such minimal memory requirements (independent of whether the underlying method is a primal-dual or simplex based algorithm). These doubts stem from the fact that a code using implicit eligibility and cost storage schemes exhibit two notable defects. First, the code is immediately problem specific. That is, as soon as the rules for eligibility or cost relevant information are changed, the code is obsolete and must be revised. Second, the code is computer dependent-- i.e., the code can only be used on one manufacturer's computer (and possibly even only one of his computer models). In the age of rapid technology and social change, it is highly doubtful that any organization should be tied to a problem specific and computer dependent solution code.

---

*To enable researchers to make meaningful comparisons of alternative solution codes we developed a computer program for generating test networks called NETGEN [19]. The NETGEN code documentation also provides the user with benchmarks (solution times on current codes and objective function values) on 40 assignment, transportation, and transshipment problems.

OUT-OF-CORE METHODS

A closely related, but even more serious misconception of the paper

concerns the application of out-of-core methods. According to [16]:

> Out-of-core approaches are considered prohibitively
> expensive. During the solution these approaches repeatedly
> access information stored on peripheral devices and can
> be shown to be impractical from the standpoint of computer
> time required. . . .
>
> It is estimated that out-of-core approaches incur a
> penalty resulting in a 10 to 1000 times increase in
> computer costs.

These speculations are contradicted by our results from testing both in-core

and out-of-core codes [17]. Based on these results, we conclude that an

appropriately designed out-of-core code is only 2 to 5 times slower than an

in-core code. This is based on the premise that the in-core code does <u>not</u>

pack information within one word. <u>If the in-core code does pack information,</u>

<u>then the out-of-core code may be faster than the in-core code.</u>

Our findings also document the following major advantages of an out-of-core

code over an in-core code:

1. Vastly larger problems can be solved by an out-of-core code. For

example, we have recently implemented an Extended Transportation System for

the U. S. Treasury Department which is capable of solving transportation

problems with 50,000 nodes and 62 million arcs on a UNIVAC 1108. Larger

problems can be solved by this system on an IBM 360/65, IBM360/155, IBM 360/165,

CDC 6600, etc.

2. Out-of-core codes require less central memory for problems of all size

ranges--including those that in-core codes can handle. This is critical for

fast job handling on multi-processing computer systems. (All of the computer

systems discussed in [16] are of this type.) Thus, if turn-around time is the

criterion of efficiency, then out-of-core codes will be substantially faster
due to the bias of multi-processing systems against jobs requiring a large
amount of core.

3. An out-of-core code can in fact operate as an in-core code simply
by allocating the code sufficient core space to bring all problem data into
core. In this mode, an appropriately designed out-of-core code runs less than
10% slower than an in-core code [17].

In conclusion, extensive research and testing has established that prac-
tical network problems can now be handled routinely at efficiencies and memory
capabilities dramatically beyond those imagined possible a few years ago. These
achievements, and the innovations that have brought them about have upended
a number of notions that unfortunately are still disseminated as folklore by
papers such as [16].

REFERENCES

1. Barr, R. S., Glover, F., and Klingman, D., "An Improved Version of the Out-of-Kilter Method and a Comparative Study of Computer Codes." To appear in *Mathematical Programming*.

2. Charnes, A. and Cooper, W. W., *Management Models and Industrial Applications of Linear Programming*, Vol. I-II (New York, John Wiley and Sons, Inc., 1961).

3. Charnes, A., Cooper, W. W., Klingman, D., and Niehaus, A., "Static and Dynamic Biased Quadratic Multi-Attribute Assignment Models: Solutions and Equivalents." CCS 115, Center for Cybernetic Studies, University of Texas, Austin, Texas.

4. Charnes, A., Glover, F., and Klingman, D., "The Lower Bounded and Partial Upper Bounded Distribution Model." *Naval Research Logistics Quarterly*, Vol. 18 (1971), pp. 277-278.

5. Charnes, A., Glover, F., and Klingman, D., "A Note On A Distribution Problem." *Operations Research*, Vol. 18, No. 6 (1970), pp. 1213-1216.

6. Charnes, A. and Klingman, D., "The Distribution Problem With Upper and Lower Bounds on the Node Requirements." *Management Science*, Vol. 16, No. 9 (1970), pp. 638-642.

6a. Ford, L. R. and Fulkerson, D , " A Primal-Dual Algorithm for the Capacitated Hitchcock Problem," *Naval Research Logistics Quarterly*, 4, 1(1957), pp. 47-54.

6b. Fulkerson, D. R., "An Out-of-Kilter Method for Solving Minimal Cost Flow Problems," *J. Soc. Indust. Appl. Math.* 9(1961), pp. 18-27.

7. Glover, F., Karney, D., and Klingman, D., "Implementation and Computational Study on Start Procedures and Basis Change Criteria for a Primal Network Code," *Networks*, Vol. 20 (1974), pp. 191-212.

8. Glover, F., Karney, D., and Klingman, D., "The Augmented Precedessor Index Method for Locating Stepping Stone Paths and Assigning Dual Prices in Distribution Problems," *Transportation Science*, Vol. 6 (1972), pp. 171-180.

9. Glover, F., Karney, D., Klingman, D., and Napier, A., "A Computational Study on Start Procedures, Basis Change Criteria, and Solution Algorithms for Transportation Problems," *Management Science*, Vol. 20, No. 5(1974), pp. 793-814.

10. Glover, F., Karney, D., and Klingman, D., "Double-Pricing Dual and Feasible Start Algorithms for the Capacitated Transportation (distribution) Problem," University of Texas at Austin (1970).

11. Glover, F., Klingman, D., and Stutz, J., "Extensions of the Augmented Predecessor Index Method to Generalized Network Problems," _Transportation Science_, Vol. 7, No. 4 (1973), pp. 377-384.

12. Glover, F. and Klingman, D., "A Note on Computational Simplification in Solving Generalized Transportation Problems," _Transportation Science_, Vol. 7 (1973), pp. 351-361.

13. Glover, F., Klingman, D., and Napier, A., "Basic Dual Feasible Solutions for a Class of Generalized Networks," _Operations Research_, Vol. 20, No. 1 (1972), pp. 126-137.

14. Glover, F., Klingman, D., and Ross, G. T., "Finding Equivalent Transportation Formulations for Constrained Transportation Problems," _Naval Research Logistics Quarterly_, Vol. 21, No. 2 (1974), pp. 247-253.

15. Glover, F., Klingman, D., and Stutz, J., "Implementation and Computational Study of a Generalized Network Code," Presented at 44th National Meeting of ORSA, San Diego, California, Nov. 12-14, 1973.

16. Hatch, Richard S. "Optimization Strategies For Large Scale Assignment and Transportation Problems," Presented at ORSA/TIMS meeting, Puerto October 16, 1974.

17. Karney, D. and Klingman, D., "Implementation and Computational Study on an In-Core Out-of-Core Primal Network Code," To appear in _Operations Research_.

18. Klingman, D., Napier, A., and Ross, G. T., "A Computational Study on the Effects of Problem Dimensions on Solution Time for Transportation Problems," Research Report CS 135, Center for Cybernetic Studies, University of Texas, Austin, Texas, (1973). To appear in _JACM_.

19. Klingman, D., Napier, A., and Stutz, J., "NETGEN - A Program for Generating Large Scale (Un) Capacitated Assignment, Transportation and Minimum Cost Flow Network Problems," _Management Science_, Vol. 20, No. 5 (1974), pp. 814-822.

20. Klingman, D., Randolph, P., and Fuller, S., "A Cotton-pickin' Cotton Ginning Problem," To appear _Operations Research_.

21. Klingman, D. and Ross, G. T., "Finding Equivalent Network Formulations For Constrained Network Problems." CS 108, Center for Cybernetic Studies, University of Texas, Austin, Texas.

22. Klingman, D. and Russell, R., "The Transportation Problem with Mixed Constraints," _Operational Research Quarterly_, 3, 4 (1974).

23. Klingman, D. and Stutz, J., "Computational Testing in Integer Generalized Network Code," Presented at 45th National ORSA/TIMS meeting, Boston, April 22-24, 1974.

24. Srinivasan, V. and Thompson, G. L., "Benefit-Cost Analysis of Coding Techniques for the Primal Transportation Algorithm," _JACM_, 20 (1973), pp. 194-213.

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author)	2a. REPORT SECURITY CLASSIFICATION
Center for Cybernetic Studies The University of Texas	Unclassified
	2b. GROUP

3 REPORT TITLE

Important Practical Misconceptions of Optimizing Large Scale Assignment and Transportation Problems

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)

Research rept.

5 AUTHOR(S) (First name, middle initial, last name)

Fred Glover
Darwin Klingman

6 REPORT DATE	7a. TOTAL NO OF PAGES	7b. NO. OF REFS
October 1974	8	24

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
N00123-74-C-2272	Center for Cybernetic Studies
b. PROJECT NO.	Research Report CCS 195
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10 DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited.

11 SUPPLEMENTARY NOTES	12 SPONSORING MILITARY ACTIVITY
	Office of Naval Research (Code 434) Washington, D.C.

13 ABSTRACT

Modelling and solving large scale networks are crucial to many practical military applications. The purpose of this note is to identify important elements of successful models and methods that were incompletely or inaccurately portrayed in recent presentations at O.R. meetings such as that in [16]. Our observations result from breakthroughs in network areas that have been validated in over a hundred computer hours of empirical testing and 15 man years of code development. They particularly apply to modelling large scale military manpower assignment problems and designing computer codes for solving large scale assignment, transportation, and transshipment problems. To focus our remarks we shall address chiefly the misconceptions presented in [16].

14 KEY WORDS	LINK A		LINK B		LINX C	
	ROLE	WT	ROLE	WT	ROLE	WT
Networks						
Assignment problems						
Linear programming						
Software development						